

This Page Is Inserted by IFW Operations
and is not a part of the Official Record

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images may include (but are not limited to):

- BLACK BORDERS
- TEXT CUT OFF AT TOP, BOTTOM OR SIDES
- FADED TEXT
- ILLEGIBLE TEXT
- SKEWED/SLANTED IMAGES
- COLORED PHOTOS
- BLACK OR VERY BLACK AND WHITE DARK PHOTOS
- GRAY SCALE DOCUMENTS

IMAGES ARE BEST AVAILABLE COPY.

**As rescanning documents *will not* correct images,
please do not report the images to the
Image Problem Mailbox.**

BNSDOCID: <WO. 9518494A1_1_>

FOR THE PURPOSES OF INFORMATION ONLY

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

AT	Austria	GB	United Kingdom	MR	Mauritania
AU	Australia	GE	Georgia	MW	Malawi
BE	Belgium	GN	Guinea	NE	Niger
BE	Belgium	GR	Greece	NL	Netherlands
BF	Burkina Faso	HU	Hungary	NO	Norway
BG	Bulgaria	IE	Ireland	NZ	New Zealand
BJ	Benin	IT	Italy	PL	Poland
BR	Brazil	JP	Japan	PT	Portugal
BY	Belarus	KE	Kenya	RO	Romania
CA	Canada	KG	Kyrgyzstan	RU	Russian Federation
CF	Central African Republic	KP	Democratic People's Republic of Korea	SD	Sudan
CG	Congo	KR	Republic of Korea	SE	Sweden
CH	Switzerland	KZ	Kazakhstan	SI	Slovenia
CI	Côte d'Ivoire	LI	Liechtenstein	SK	Slovakia
CM	Cameroon	LK	Sri Lanka	SN	Senegal
CN	China	LU	Luxembourg	TD	Chad
CS	Czechoslovakia	LV	Latvia	TG	Togo
CZ	Czech Republic	MC	Monaco	TJ	Tajikistan
DE	Germany	MD	Republic of Moldova	TT	Trinidad and Tobago
DK	Denmark	MG	Madagascar	UA	Ukraine
ES	Spain	ML	Mali	US	United States of America
FI	Finland	MN	Mongolia	UZ	Uzbekistan
FR	France			VN	Viet Nam
GA	Gabon				

DATA FRAME FORMAT FOR VARIABLE SIZE DATA CONSTELLATIONS

This invention relates generally to digital transmission systems and particularly to a digital data transmission system having a data frame structure and circuit arrangement selected to facilitate receiver operations such as symbol to byte conversion, deinterleaving and forward error correction and a data rate that is related to the signal to noise ratio (S/N ratio) of the transmission environment for enhancing system capacity.

U.S. Patent No. 5,087,975 discloses a vestigial sideband (VSB) system for transmitting a television signal in the form of successive M-level symbols over a standard 6 MHz television channel. The television signal may, for example, comprise one or two compressed wideband HDTV signals or a number of compressed NTSC signals. While the number of levels M characterizing the symbols may vary depending on circumstances, the symbol rate is preferably fixed, such as at a rate of 684 H (about 10.76 Megasymbols/sec), where H is the NTSC horizontal scanning frequency. The number of symbol levels used in any particular situation is largely a function of the S/N ratio characterizing the transmission medium, a smaller number of symbol levels being used in situations where the S/N ratio is low. It is believed that the ability to accommodate symbol levels of 24, 16, 8, 4 and 2 provides adequate flexibility to satisfy conditions in most systems. It will be appreciated that lower values of M can provide improved S/N ratio performance at the expense of reduced transmission bit rate. For example, assuming a rate of 10.76 M symbol/sec, a 2-level VSB signal (1 bit per symbol) provides a transmission bit rate of 10.76 Megabits/sec, a 4-level VSB signal (2 bits per symbol) provides a transmission bit rate of 21.52 Megabits/sec and so on up to a 24-level VSB signal which provides a transmission bit rate of about 48.43 Megabits/sec.

It is generally known that the S/N ratio performance of cable television plants decreases as the signal (channel) frequency increases. The foregoing attribute of an M-level VSB transmission system, i.e. improved S/N ratio performance as M decreases, is used in one aspect of the invention to

- 2 -

compensate for the S/N ratio degradation in the higher frequency channels of CATV distribution plants. That is, according to this aspect of the invention, VSB transmission is effected in a CATV system wherein the lower frequency channels are transmitted using larger values of M and the higher frequency channels are transmitted using lower values of M. While the bit rate of the higher frequency channels is thereby reduced, the received signal may be reproduced with a S/N ratio comparable to that of the lower frequency channels.

Moreover, in accordance with other aspects of the invention, system efficiency, particularly in relation to receiver operations such as data deinterleaving, symbol to byte conversion and forward error correction, may be greatly enhanced by selecting a data frame structure which facilitates these operations within the constraints of the variable M-level VSB character of the transmitted signal.

Accordingly, a principal object of this invention is to provide a novel data format for digital data transmission.

Another object of the invention is to provide a novel data frame structure useful in the transmission of a digital information signal having a variable data constellation.

These and other objects and advantages of the invention will be apparent upon reading the following description in conjunction with the drawings, in which:

FIG. 1 illustrates the novel data frame structure of the invention;

FIG. 2 is a chart showing the relationship of data constellation size to the other parameters of the invention;

FIG. 3 is a simplified block diagram of a transmitter in accordance with the invention;

FIG. 3A is a chart illustrating an implementation of the byte to symbol converter in the transmitter of FIG. 3;

FIG. 3B is a chart illustrating the variation of S/N ratio with frequency in a cable plant and assignment of VSB mode for optimizing cable plant operation;

- 3 -

FIG. 4 is a simplified diagram of a receiver constructed in accordance with the invention;

FIG. 5 is a more detailed showing of the data processor section of the receiver of FIG. 4;

FIG. 6 is a more detailed showing of the symbol to byte converter of the data processor of FIG. 5;

FIG. 7 is a diagram of an exemplary simple linear memory array for deinterleaving a convolutionally interleaved data stream;

FIG. 8 is a generalized circuit for generating the memory addresses for a memory array such as that illustrated in FIG. 7;

FIG. 9 is a memory array of a ROM suitable for use in an actual embodiment of the invention; and

FIG. 10 is a replacement circuit for the ROM of FIG. 9.

The structure of the novel data frame of the invention is illustrated in FIG. 1. The data frame format is the subject of this invention. The data frame, generally identified by reference numeral 10, comprises 313 data segments (DS) identified as DS0-DS312. Each data segment includes 832 symbols for data and 4 symbols defining a data segment sync character 11. Each data segment sync character 11 comprises four 2-level symbols which preferably assume the form disclosed in copending application Serial No. 894,388. The first data segment DS0 comprises only 2-level symbols. These 2-level symbols represent a pseudo random sequence frame synchronization code, and an 8 symbol VSB mode, or control, signal that identifies the level M (e.g. 24, 16, 8, 4 or 2) of the symbols of the data field of the remaining 312 data segments of the data frame. The first three bits of the eight bit VSB mode control byte identify the VSB mode and the remaining 5 bits comprise parity bits which may be used by the receiver for error detection. In the previously mentioned CATV distribution system, for example, the VSB mode signal would identify a relatively large M for low frequency channels

- 4 -

and a smaller M for higher frequency channels. In a terrestrial broadcast environment, the VSB mode would most likely be M=2, 4 or 8.

Referring to the table of FIG. 2, each data symbol of a data segment DS1-DS312 represents either 4.5 bits (M=24), 4 bits (M=16), 3 bits (M=8), 2 bits (M=4), or 1 bit (M=2). Since there are a fixed number of data symbols per frame ($312 \times 832 = 259,584$), the number of data bytes per frame will vary as shown. That is, each frame comprises 146,016 data bytes for VSB mode M=24; 129,792 data bytes for VSB mode M=16; 97,344 data bytes for VSB mode M=8; 64,896 data bytes for VSB mode M=4; and 32,448 data bytes for VSB mode M=2. However, while the number of data bytes per frame varies depending on the VSB mode M, it will be observed that for any particular value of M (24, 16, 8, 4 or 2), an integral number of bytes is provided in each frame. This characteristic of the structure of frame 10 substantially simplifies the design of a receiver such that frame synchronization will be effected at a rate corresponding to an integral number of data bytes regardless of the VSB mode (i.e. 24, 16, 8, 4 or 2 level symbols). As will be explained in further detail hereinafter, the receiver forward error correction circuitry, the receiver symbol to byte converter, and the receiver deinterleaver are preferably frame synchronized with the transmitted signal. The frame sync signal can be directly used for these purposes so long as there are an integral number of bytes, forward error correction blocks and interleave blocks in each data frame for each of the VSB modes.

Reed-Solomon (RS) forward error correction is used in the receiver of the invention. A standard transport packet size of 188 bytes has been established by the MPEG (Motion Picture Experts Group) committee. Adding 20 parity bytes to each such transport packet results in an RS block size of 208 data bytes, allowing for the correction of 10 byte errors per RS block. As seen in FIG. 2, an RS block size of 208 bytes advantageously results in an integral number of RS blocks per

- 5 -

frame for all of the selected VSB modes, thereby allowing the receiver RS decoder to be synchronized by the frame sync signal. An interleave block (I block) is defined according to the invention as comprising 26 RS blocks (5,408 data bytes) which also results in an integral number of I blocks per frame regardless of the selected VSB mode (see FIG. 2), thereby also allowing the frame sync signal to be used to periodically synchronize the receiver deinterleaver in the case where a block interleave algorithm is employed (data stored in row format and read out in column format). For convolutional interleaving, the size of a parameter B (to be defined hereinafter) must be selected such that each frame (regardless of VSB mode) comprises an integral number of B data bytes, thereby allowing for frame synchronization. In a preferred embodiment of the invention B=26 data bytes. Also, the use of an integral number of data bytes per frame, regardless of VSB mode, enables direct frame synchronization of the receiver symbol to byte converter.

FIG. 3 is a simplified block diagram of a transmitter constructed in accordance with the invention. A source 12 of television signals is coupled to a convolutional interleaver 13 which, in turn, supplies interleaved data bytes to a byte to symbol converter and mapper 14. It will be appreciated that signal source 12 may comprise a compressed HDTV signal (or two compressed HDTV signals depending on the VSB mode) or a number of compressed NTSC signals. The symbol output of converter (and mapper) 14 is supplied to a frame formatter 15 that has inputs of data, data segment sync and frame sync and is controlled, along with converter 14, by a VSB mode control signal. The formatted frames, which conform to the arrangement previously described in connection with FIGS. 1 and 2, are supplied to a digital to analog (D/A) converter 16 and thence to a VSB transmitter 17 for transmission over a 6 MHz television channel. The transmission medium may comprise a cable television plant or a terrestrial broadcast environment. In either case, one such

- 6 -

transmitter is required for each transmitted 6 MHz channel.

FIG. 3A is a chart illustrating a method of implementing the byte to symbol converter. The chart comprises four columns, one for each of the VSB modes $M=16$, $M=8$, $M=4$ and $M=2$. Converter 14 is operative in response to the applied VSB mode control signal for using the identified column of the chart of FIG. 3A for converting the input data bytes to output data symbols. For example, for VSB mode $M=16$, the input data byte 11010101 would be converted to two successive data symbols having relative amplitudes of +88 and -40. For VSB mode $M=8$, the input data byte would be converted to three successive data symbols having relative amplitudes of +80, +48 and -16 (assuming the first bit of the next data byte is 1) or +80, +48 and -48 (assuming the first bit of the next data byte is 0). For VSB mode $M=4$, the data byte would be converted to four successive symbols having relative amplitudes of +96, -32, -32 and -32. Finally, for VSB mode $M=2$, eight output symbols would be provided at relative amplitudes +64, +64, -64, +64, -64, +64, -64 and +64. In the case of VSB mode $M=24$, converter and mapper 14 includes appropriate memory and look-up tables for mapping successive groups of 9 input data bits into two successive 24 level output symbols. In this way each symbol can be referred to as representing 4.5 bits (i.e. 9 bits/2 symbols).

In connection with the foregoing, it will be observed that the relative levels of the symbols of each VSB mode are evenly spaced and lie midway between the relative levels of selected symbols of all higher VSB modes. For example, relative level +112 of VSB mode $M=8$ lies midway between relative levels +120 and +104 of VSB mode $M=16$, relative level +96 lies midway between relative levels +112 and +80 of VSB mode $M=8$ and midway between relative levels +104 and +88 of VSB mode $M=16$, relative level +64 of VSB mode $M=2$ lies midway between relative levels +96 and +32 of VSB mode $M=4$, +80 and +48 of VSB mode $M=8$ and +72 and +56 of VSB mode $M=16$, and so on. Preferably the symbol levels are offset

- 7 -

from the values shown by a predetermined amount (e.g. +20) prior to transmission to provide a small pilot for facilitating carrier acquisition in the receiver. Also, it will be observed that the data rate characterizing each VSB mode increases by one bit per symbol relative to the data rate of the immediately lower VSB mode while its S/N ratio performance is reduced by one-half.

The above represents the situation in the version of the invention presently being implemented in chip form, in which 16 VSB is the highest mode. However, it will be apparent to those skilled in the art that the 24 VSB mode implementation will require different relative levels as taught by the invention. The technique and apparatus for processing a 24 VSB mode signal is discussed below and claimed in copending application D-6718, above.

FIG. 3B illustrates the general S/N ratio fall off of a cable plant with increasing frequency. As is specified in the invention, the higher S/N ratio transmission characteristics for lower frequency television signals can be used to advantageously employ higher VSB transmissions (and a higher data rate) for similar performance. The "noisier" portion of the spectrum (low S/N ratio) may be used for signals of lower VSB modes. It will be apparent that this technique optimizes the utilization of the cable plant.

FIG. 4 is a simplified block diagram of a receiver constructed according to the invention. The received RF television signal from the transmitter of FIG. 3 comprises an M-level VSB signal having the frame format of FIG. 1. The received signal is converted to an IF frequency by a tuner 20 and applied to a VSB demodulator 22. VSB demodulator 22 generates an analog baseband output signal comprising the M-level symbols at a rate of about 10.76 Megasymbols/sec. This analog signal is sampled by an analog to digital (A/D) converter 24 which converts the symbols to binary form and applies them to a data processor 26. Data processor 26 provides a feedback signal for controlling A/D 24 to insure

- 8 -

that the analog baseband signal is sampled at the appropriate symbol times (see copending application Serial No. 894,388). Data processor 26 applies the processed binary data, in the form of data bytes corresponding to the output of television source 12 in FIG. 3, to a demultiplexer 28, which distributes the received data to a video processor 30 and to an audio processor 32, each of which includes appropriate decompression circuitry.

Data processor 26 is shown in more detail in FIG. 5. The binary symbols from A/D 24 are applied to a data acquisition circuit 34 which performs a number of functions including generating the feedback signal for controlling A/D 24, generating a symbol clock signal, generating a frame sync (FSYNC) signal and generating an 8 times symbol clock signal. The symbol clock signal has a frequency of about 10.76 MHz for all VSB modes except VSB mode $M=24$, in which case, as will be explained, its frequency is reduced by one-half to about 5.38 MHz. The FSYNC used in the preferred embodiment is approximately 41 Hz. The frame synchronization code of data segment DS0 enables derivation of the FSYNC signal which coincides in time with the first data symbol of data segment DS1 of each data frame 10.

The binary symbols from A/D 24 (representing the amplitudes of the sampled analog signal from VSB demodulator 22) are applied by data acquisition circuit 34 to a multi-level slicer 36 which converts the received symbols to bits according to the chart of FIG. 3A. As in the case of the transmitter, for VSB mode $M=24$, slicer 36 includes appropriate memory and look-up tables for mapping each successively received pair of symbols into 9 bits of output data. Slicer 36 couples the sliced values of the VSB mode control byte (eight 2-level symbols) of data segment DS0 of each frame to a VSB mode decoder 37, which detects the first three bits of the mode control byte and develops a 3-bit VSB mode select signal. This signal identifies the VSB mode ($M=24, 16, 8, 4$ or 2) of the received symbols for controlling data acquisition circuit

- 9 -

34, slicer 36 and a symbol to byte converter 38 during the remainder of the respective frame. The remaining 5 bits of the mode control byte are used for error detection by VSB mode decoder 37. Slicer 36, which includes a nine line output bus, is responsive to the VSB mode select signal for converting the binary signal, representing the symbol amplitudes, to their corresponding bit values. Thus, in the M=2 VSB mode, each binary symbol amplitude signal is converted to the corresponding 1-bit signal on one of the nine output lines, to the corresponding 2-bit signal on two of the output lines in the M=4 VSB mode, to the corresponding 3-bit signal on three of the output lines in the M=8 VSB mode and to the corresponding 4-bit signal on four of the output lines in the M=16 VSB mode. In the M=24 VSB mode, two successive symbols are converted to the corresponding 9-bit signal (on all nine output lines) at one-half the symbol clock rate.

The nine-line output of slicer 36, together with the 3-bit VSB mode select signal from decoder 37 and the timing signals from data acquisition circuit 34 are coupled to symbol to byte converter 38, the output of which supplies a deinterleaver 40 that, in turn, supplies an RS decoder 42. Symbol to byte converter 38 converts the input bits representing the received symbols into a series of 8-bit data bytes for each of the VSB modes. Deinterleaver 40 deinterleaves the convolutionally interleaved data bytes supplied by converter 38 and RS decoder 42 performs error correction on the deinterleaved data bytes.

Symbol to byte converter 38 is shown in more detail in FIG. 6. The 9-line data symbol bus from slicer 36 and the symbol clock signal from data acquisition circuit 34 are applied to a 1 x 9 parallel-in, serial-out register 44. The 3-bit VSB mode signal, the symbol clock signal and the 8 times symbol clock signal are applied to the inputs of a variable pulse generator 46. Generator 46 produces a series of pulses (bit clocks) at the 8 times symbol clock rate in response to each received symbol clock, the series including one pulse for

- 10 -

VSB mode $M=2$, two pulses for VSB mode $M=4$, 3 pulses for VSB mode $M=8$, and 4 pulses for VSB mode $M=16$. A series of 9 pulses are sent for VSB mode $M=24$, while the symbol clock is halved.

Assuming, for example, VSB mode $M=8$, the 3 bits from slicer 36 (representing a received symbol) are simultaneously written into the top three locations of register 44 in response to a symbol clock. At the same time, the symbol clock starts pulse generator 46 which produces 3 bit clocks (at 8 times the symbol clock rate) which are applied to the Read (RD) input of register 44, to the Write (WR) input of a 1 x 8 Serial-in, Parallel-out register 48 and to the input of a divide-by-eight circuit divider 50. Divider 50 is reset at the beginning of each data frame 10 by FSYNC for synchronizing converter 38 with the received data, since FSYNC always represents a byte boundary. The 3 bits previously loaded into register 44 are serially read out of register 44 in response to the 3 bit clock pulses and written into register 48. The next 3-bit symbol from slicer 36 is processed in the same manner, resulting in 6 bits being stored in register 48. After the first two bits of the next (i.e. third) 3-bit symbol are written into register 48 from register 44, divider 50 generates an output (Byte Clock) causing the accumulated 8-bit data byte stored in register 48 to be read out (in parallel). The remaining single bit of the third symbol is stored in register 48 and read out with the next 7 bits of the succeeding 3-bit symbol as the next data byte. In this manner, converter 38 arranges the input 3-bit symbols (for $M=8$) into a series of successive 8-bit output data bytes, which are supplied to deinterleaver 40 at the rate of the byte clock.

A substantially similar procedure is carried out for VSB modes $M=2$, 4 and 16 except that, depending on the VSB mode, it will take a longer or shorter time period to compose the data bytes. For example, in VSB modes $M=2$ and 4, the bit clock generated by pulse generator 46 in response to each

- 11 -

symbol clock signal comprises one and two pulses respectively, so that 8 and 4 symbol clock signals are required to produce the respective data bytes (versus $2\frac{2}{3}$ for VSB mode $M=8$).

For VSB mode $M=24$, slicer 36 combines two 4.5 bit symbols into 9 bits at one-half the symbol clock rate and applies these signals to converter 38. Consequently, in response to each symbol clock, 9 bits from slicer 36 are loaded into register 44. Then, the 9 bits are serially read out of register 44 and written into register 48. However, after the 8th such bit has been written into the register, a Byte Clock output is generated by divider 50, reading these 8 bits out of register 48 as a data byte. The remaining bit will be read out of register 48 with the first 7 bits of the next symbol as the next data byte two symbol times later.

Converter 38 further includes a divider 52 which is also reset by FSYNC to synchronize its operation with the beginning of each data frame 10. In particular, divider 52 is a divide-by-208 counter which is clocked by the Byte Clock generated by divider 50 to generate a Begin RS Block signal for identifying the beginning of each block of 208 data bytes synchronously with the first data byte of each data frame. As will be explained hereinafter, this signal is used to synchronize the operation of RS decoder 42. Divider 52 includes means responsive to the symbol and byte clocks for converting the one-symbol-wide input FSYNC to a one byte wide output PSYNC.

As described earlier, the outputs of converter 38 are applied to deinterleaver 40. As will be explained in further detail hereinafter, deinterleaver 40 deinterleaves the convolutionally interleaved data bytes received from converter 38 using a minimum of memory.

As is well known, interleaving is done at the transmitter (see interleaver 13 of FIG. 3) to spread contiguous data bytes apart from each other to help immunize the transmitted data from burst noise. In the receiver, the

- 12 -

interleaved bytes must be deinterleaved to re-establish their original relationship prior to forward error correction. Thus, burst noise of some given time duration will corrupt only a limited number of bytes within an RS block of the deinterleaved data, which corrupted bytes can be corrected by the RS decoder. The interleaving algorithm used is selected in anticipation of the maximum expected burst noise duration at the fastest byte clock rate (i.e. corresponding to VSB mode $M=24$) to insure that the RS decoder is capable of error correcting the corrupted deinterleaved data bytes. Thus, as maximum expected burst noise duration increases, the interleaving algorithm must spread contiguous data bytes farther apart. Alternatively, a more powerful RS code may be used, but this approach has the disadvantage of using more overhead, i.e. requiring more bytes for error correction. Also, by referencing the system to the highest byte clock rate (corresponding to 24 VSB), increased burst error protection will be provided as the VSB mode and the corresponding byte rate decrease, because the interleave pattern is effected over a given number of bytes regardless of VSB mode.

Convolutional interleave algorithms are commonly used to immunize transmitted data from burst noise. Such algorithms delay the individual bytes of successive groups of bytes by different amounts to effectively scatter the bytes over a portion or all of the data frame. Deinterleaving is effected by delaying the received bytes by opposite amounts. In implementing such a system, three parameters are of particular significance; the maximum expected burst length BL , the number of byte errors T which the RS decoder can correct and the RS block size N . As mentioned previously, there preferably are an integral number of RS blocks in the data frame so that the RS decoder can be synchronized by the frame sync signal $FSYNC$. By selecting an interleave block size (of which there are preferably an integral number in each frame) equal to a parameter $B=BL/T$ and the different delays as integral multiples of the RS block size N , the RS decoder will

- 13 -

be able to correct the deinterleaved data for burst noise up to the maximum expected duration of BL byte clocks.

Consider the simplified example of a system in which the maximum expected burst length is 4 data byte clocks and the RS decoder is capable of correcting one data byte error in each 8 data byte RS block (i.e. $BL=4$, $T=1$, $N=8$). Using these parameters, the interleave block size $B=BL/T=4/1=4$. Convolutional interleaving is performed using these parameters such that, for each group of $B=4$ data bytes, the first data byte is exposed to a delay of 0, the second to a delay of $1N=8$ data byte clocks, the third to a delay of $2N=16$ data byte clocks and the fourth to a delay of $3N=24$ data byte clocks. Deinterleaving is effected by reversing the delays such that for each group of $B=4$ received interleaved data bytes, the first is delayed by $3N=24$ data byte clocks, the second by $2N=16$ data byte clocks, the third by $1N=8$ data byte clocks and the fourth by 0.

Conventional convolutional deinterleavers implementing the above algorithm comprise a memory having $(B-1)N/2$ memory locations. For realistic values of B and N , which are typically much larger than the values used in the simplified example given above, this leads to an impractical architecture because of the large number of shift registers required. An alternate architecture which may be employed uses a standard linear memory array for which a large number of fifo head and tail pointers must be maintained in hardware. This is a very complex task and thus highly undesirable.

These problems are solved according to the invention by using a linear memory array with an address generator for generating a repeating sequence of read-write addresses that results in correctly deinterleaving the received data. The memory array is of a relatively small size utilizing only one memory location in excess of the number required to impose the different delays on the respective data bytes of each group. This invention is described below in relation to the simplified example given above and is extended to an

- 14 -

implementation using more realistic parameters. This aspect of the invention is separately claimed in copending application D-6727, above.

It has been discovered that a linear memory array having $[(B-1)N/2+1]$ memory locations may be addressed by a particular sequence of read-write addresses to correctly deinterleave a convolutionally interleaved data stream. The sequence of addresses is developed by first writing a column of $[(B-1)N/2+1]$ consecutive integers beginning with 0. This is represented by the first column of the addressing matrix of FIG. 7, which is for the simplified example with $B=4$ and $N=8$. Next, a second column that is the same as the first is written except that it is rotated down by $(B-1)N/B=(3 \times 2)=6$ rows. This is the second column in the matrix of FIG. 7. The third column is developed by rotating the second column down by $(B-2)N/B=(2 \times 2)=4$ rows and, finally, the fourth and last column is developed by rotating the third column down by $(B-3)N/B=(1 \times 2)=2$ rows. This matrix has a size of $B=4$ columns and $[(B-1)N/2+1]=13$ rows, the number of rows being the required size of the deinterleave linear memory array. If the numbers in the matrix are read out a row at a time, they can be used as read-write addresses for application to the deinterleave memory to continuously deinterleave the received data, i.e. read out the old data at the addressed memory location and write in the new data at the same location.

A circuit for generating the address matrix of FIG. 7 is illustrated in FIG. 8. A divide by B counter 60, which is reset by FSYNC, is responsive to the byte clock from converter 38 for applying a repeating sequence of counts from a minimum number to a maximum number, e.g. from 0 to $(B-1)$, for addressing a B word ROM 62 at the byte clock rate. ROM 62 stores the B words comprising the first row of the address matrix. The carry output of divider 60 is used to clock a second divider 64 at the slower rate of byte clock/B, the division ratio of divider 64 being $((B-1)N/2)+1$. The outputs of divider 64 and ROM 62 are summed

- 15 -

in an adder 66 and applied through a modulo $[(B-1)N/2+1]$ converter for addressing a linear deinterleave memory array 70 having

$((B-1)N/2)+1$ memory locations. Converter 68 preferably comprises a circuit which either couples its input signal to its output or, if the input is greater than $((B-1)N/2)+1$, subtracts this value from the input signal before coupling it to its output.

For the simplified example described above, i.e. $B=4$ and $N=8$, divider 60 (+4) repetitively generates the output count sequence 0, 1, 2, 3 for addressing and reading the four words 0, 7, 3, 1 stored in ROM 62 at the byte clock rate. Divider 64 (+13) repetitively generates the output sequence 0, 1, 2 ... 12 at $1/4$ the byte clock rate. Consequently, for each group of 4 successive byte clocks, the four words 0, 7, 3, 1 read from ROM 62 are each added to the output of divider 64 in adder 66 to produce, after processing by modulo 13 converter 68, a row of the address matrix of FIG. 7. For example, the first row 0, 7, 3, 1 of the matrix is produced during the first four byte clocks wherein 0 (from divider 64) is added to the outputs 0, 7, 3, 1 of ROM 62. During the next 4 byte clocks the value 1 (from divider 64) is added to the words 0, 7, 3, 1 generated by ROM 62 to produce the second row 1, 8, 4, 2 of the address matrix and so on until finally the last row is generated by adding 12 to each of the 4 words 0, 7, 3, 1 to produce the last row 12, 6 (=19 modulo 13), 2 (=15 modulo 13), 0 (=0 modulo 13).

Each address signal generated at the output of modulo 13 converter 68 initiates a read-write operation of RAM 70. That is, each address first causes the data byte stored at the addressed memory location to be read out on bus 72 and then the new interleaved data byte to be written into the same memory location. By following the addressing scheme disclosed herein, the data bytes read from RAM 70 will be in the correct deinterleaved format. The fact that the input data bytes are correctly deinterleaved can be seen from a close examination

- 16 -

of the matrix of FIG. 7. In particular, it will be recalled that convolutional interleaving is effected for each group of four data bytes by delaying the first data byte by 0, the second data byte by 8 byte clocks, the third data byte by 16 byte clocks and the fourth data byte by 24 byte clocks. Theoretically, deinterleaving is then effected by delaying the first received data byte by 24 byte clocks, the second received data byte by 16 byte clocks, the third received data byte by 8 byte clocks and the fourth received data byte by 0 byte clocks. Adding a one byte clock delay to each received data byte will not affect the deinterleaving process so that delays of 25, 17, 9, 1 data clocks are equivalent to using deinterleaving delays of 24, 16, 8, 0 data clocks. Referring to the matrix of FIG. 7, the first received data byte is written into memory location 0 of RAM 70 and read out 25 byte clocks later at column 2, row 7 (i.e. after a delay of 25 byte clocks), the second is written into memory location 7 and read out 17 byte clocks later at column 3, row 5, (i.e. after a delay of 17 byte clocks), the third is written into memory location 3 and read out 9 byte clocks later at column 4, row 3 (i.e. after a delay of 9 byte clocks) and the fourth is written into memory location 1 from which it is read out 1 byte clock later at column 1, row 2 (i.e. after a delay of 1 byte clock). It will be apparent from this analysis that the deinterleaving algorithm is correctly executed.

In an embodiment of the invention actually being constructed, $BL=260$ bytes, $T=10$ and $N=208$. Therefore, $B=BL/T=26$ bytes, which is integrally related to the frame size for all VSB modes. In the circuit of FIG. 8, divider 60 is a divide-by-26 divider (periodically reset by FSYNC) which sequentially address the 26 words of ROM 62 comprising the first row of the address matrix (0, 2401, ... 1) shown in FIG. 9. Divider 64 is a divide by 2601 divider and converter 68 is a modulo 2601 converter. Operation of this embodiment of the circuit is identical to that described for the simplified example.

- 17 -

It will be noted that the circuit of FIG. 8 automatically adapts to the different received VSB modes. The circuit simply operates at an increased or decreased rate in response to the byte clock as the VSB mode varies. The deinterleaved data on bus 72, together with the FSYNC and Begin RS Block signals from converter 38 are applied to RS decoder 42 which error corrects the deinterleaved data bytes. Operation of RS decoder 42 is facilitated by synchronization with FSYNC at the beginning of each data frame and by providing an integral number of RS blocks each frame regardless of the VSB mode as previously explained.

FIG. 10 shows a circuit which may be used in place of ROM 62 of FIG. 8 to generate the first row of matrix addresses. The circuit implements the following equations for generating the first row of addresses $C(X)$:

$$C(0)=0$$

$$C(X)=[C(X-1)-(B-X)N/B] \bmod [(B-1)N/2 + 1],$$

where $X=1, \dots (B-1)$.

For the simplified example (i.e. $B=4$ and $N=8$) the term $C(X)$ becomes $C(X)=[C(X-1)-(4-X)2] \bmod 13$. Thus, for example, if $X=1$, $C(X)=(0-6) \bmod 13=-6 \bmod 13=7$. Similarly, for $X=2$, $C(X)=(7-4) \bmod 13=3$. And finally, for $X=3$, $C(X)=(3-2) \bmod 13=1$. The first row for the larger matrix may likewise be derived using these relationships. The circuit comprises an input divide by B count down divider 80 which is reset by FSYNC and clocked by the byte clock. In response to being reset, an output of divider 80 clears a latch 82. The output of divider 80 is provided in reverse order ($B-1, B-2, \dots 0$) so that when it is incremented by one (1) in an adder 84, the result is $(B-X)$. The variable $(B-X)$ is multiplied by N/B in a multiplier 86 to provide the expression $(B-X)N/B$. This is subtracted from the output $C(X-1)$ of latch 82 in subtractor 88 to provide $C(X-1)-(B-X)N/B$. Finally, the output of the subtractor is coupled by a modulo $[(B-1)N/2+1]$ circuit for temporary storage in latch 82 in response to a byte clock.

- 18 -

The following claims are directed to the aspect of the invention involving the data frame format for the variable size data constellations. It will be appreciated that this invention is to be limited only as defined in the claims.

- 19 -

CLAIMS

1. A method of formatting a data frame comprising providing a given number DS of repetitive data segments, each of said data segments containing a fixed number S of M-level symbols representing a plurality of data bytes and a data segment sync character, and choosing M, S and DS such that said data frame comprises an integral number of said data bytes.
2. The method of claim 1, including the step of defining an error correction block as comprising a predetermined number of said data bytes and wherein the chosen values result in said data frame comprising an integral number of said error correction blocks.
3. The method of claim 1 or 2, wherein M comprises one of a selected plurality of integer values.
4. The method of claim 3, wherein said plurality of integer values comprise at least 2, 4, 8 and 16.
5. The method of claim 3, wherein one of said data segments comprises a header segment including a constellation size code defining the chosen value of M for said frame and a frame sync code.
6. The method of claim 5, wherein said header segment constitutes the first data segment of said frame and wherein said frame sync code and said constellation size code comprise two level symbols.
7. A data receiver including means for receiving an information signal comprising a plurality of repetitive frames each including a given number DS of repetitive data segments, each of said data segments containing a fixed number S of M-level symbols representing a plurality of data bytes and a data segment sync character, DS, S and M being related such that each of said data frames comprises an integral number of said data bytes, means for detecting and quantizing said received information signal, and means for processing said detected and quantized information signal.

- 20 -

8. The receiver of claim 7, including means responsive to said information signal for generating a reset signal at the beginning of each of said frames, said processing means being reset at the beginning of each of said frames in response to said reset signal.
9. The receiver of claim 8, wherein each of said frames of said information signal comprises an integral number of error correction blocks and wherein said processing means comprises a forward error correction circuit for error correcting said error correction blocks and which is reset at the beginning of each of said frames in response to said reset signal.
10. The receiver of claim 7, 8 or 9, wherein M comprises one of a selected plurality of integer values.
11. The receiver of claim 10, wherein said plurality of integer values comprise at least 2, 4, 8 and 16.
12. The receiver of claim 7, wherein one of said data segments comprises a header segment including a constellation size code defining the chosen value of M for said frame and a frame sync code.
13. The receiver of claim 12, wherein said header segment constitutes the first data segment of said frame and wherein said frame sync code and said constellation size code comprise two level symbols.

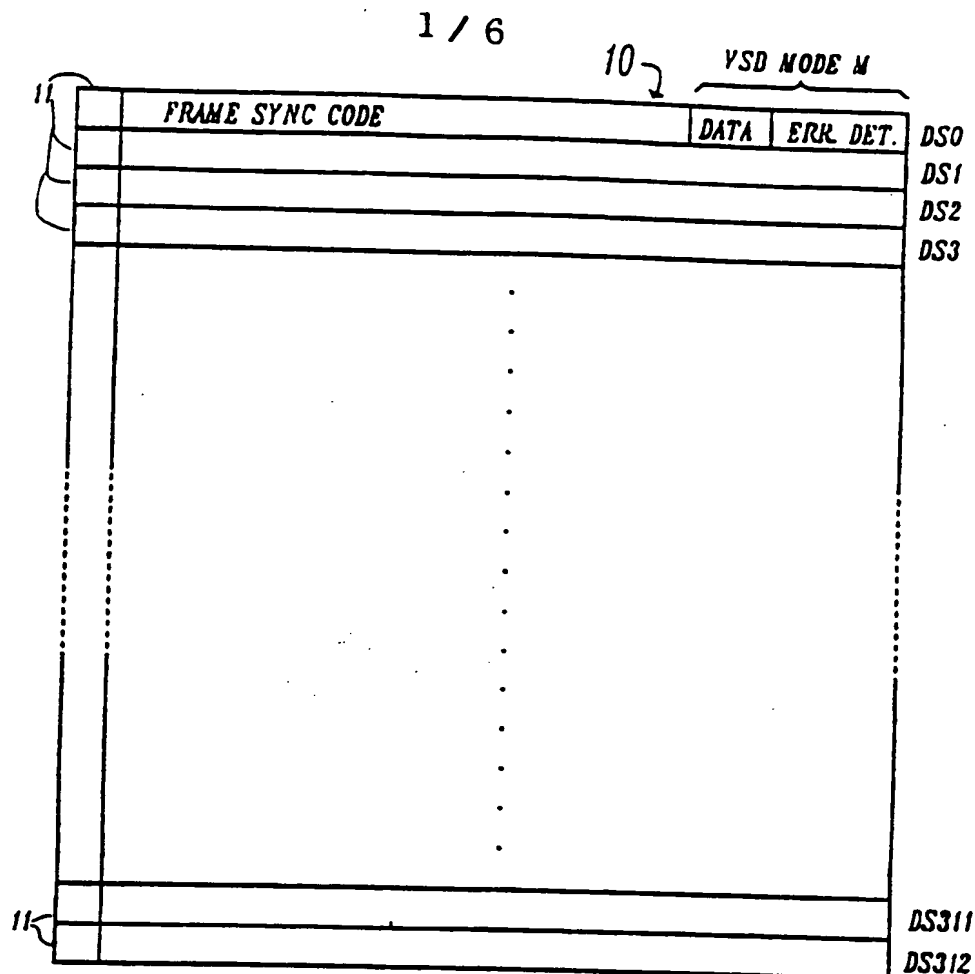


FIG. 2

VSD MODE M	DATA SYMBOLS PER FRAME	DITS PER SYMBOL	DATA BYTES PER FRAME	RS BLOCKS PER FRAME	INTERLEAVE BLOCKS PER FRAME
24	259,584	4.5	146,016	702	5616
16	259,584	4	129,792	624	4992
8	259,584	3	97,344	468	3744
4	259,584	2	64,896	312	2496
2	259,584	1	32,448	156	1248

2 / 6

FIG. 3

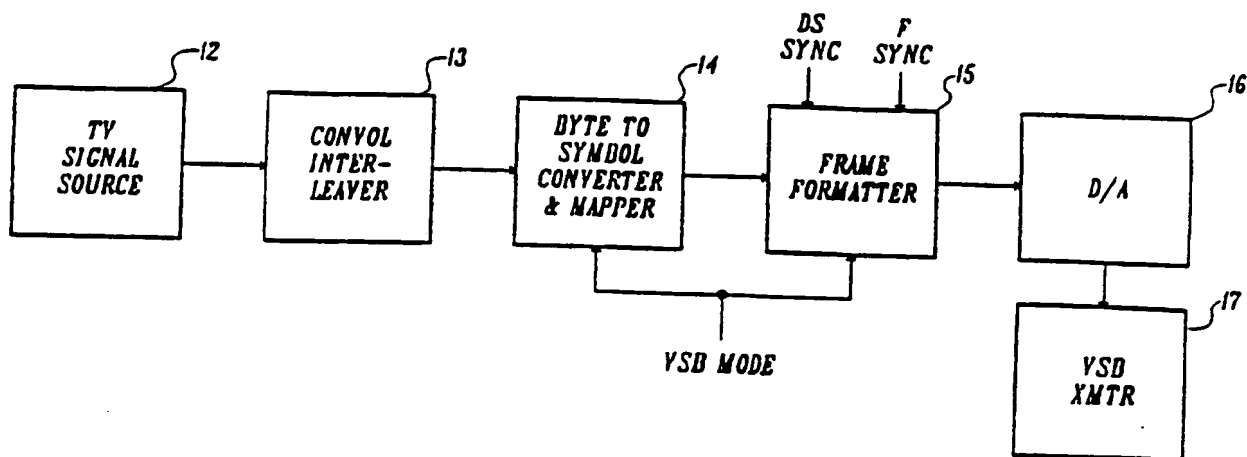


FIG. 4

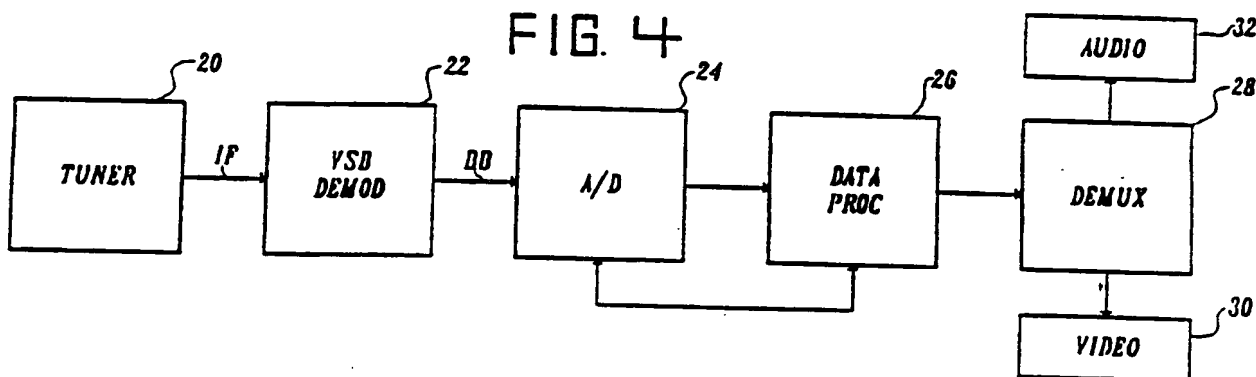
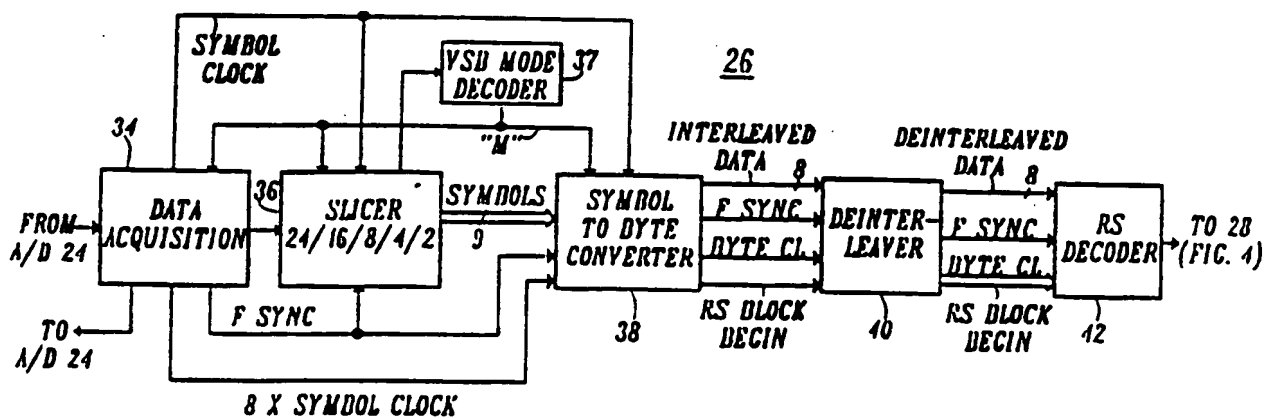


FIG. 5



3 / 6

VSD MODE M=16

VSD MODE M=8

VSD MODE M=4

VSD MODE M=2

<u>1111</u> +120	<u>111</u> +112		
<u>1110</u> +104		<u>11</u> +96	
<u>1101</u> +88	<u>110</u> +80		
<u>1100</u> +72			<u>1</u> +64
<u>1011</u> +56	<u>101</u> +48		
<u>1010</u> +40		<u>10</u> +32	
<u>1001</u> +24	<u>100</u> +16		
<u>1000</u> +8			
<u>0111</u> -8	<u>011</u> -16		
<u>0110</u> -24		<u>01</u> -32	
<u>0101</u> -40	<u>010</u> -48		
<u>0100</u> -56			<u>0</u> -64
<u>0011</u> -72	<u>001</u> -80		
<u>0010</u> -88		<u>00</u> -96	
<u>0001</u> -104	<u>000</u> -112		
<u>0000</u> -120			

FIG. 3A

4 / 6

FIG. 3B

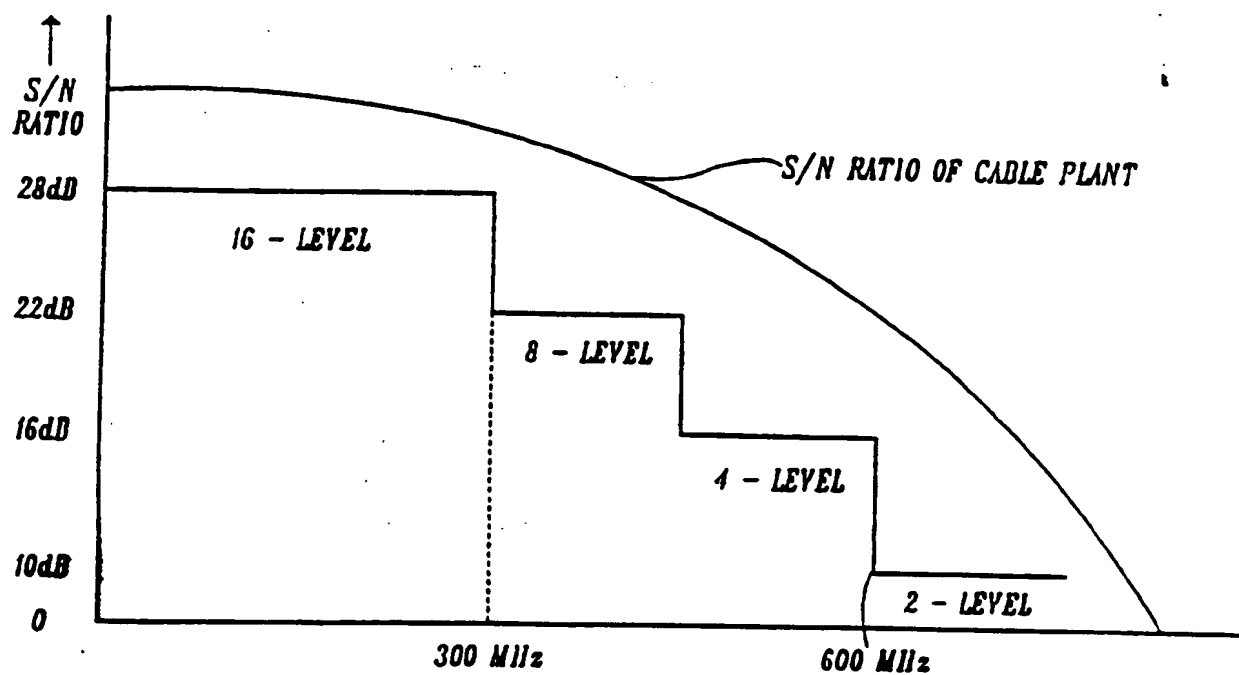


FIG. 6

Block diagram illustrating a digital system for processing data symbols, showing the following components and connections:

- 1x8 SER IN PAR OUT (18):** A register with **WR** (Write) and **RD** (Read) ports. It outputs **8 DATA** and receives **BYTE CLOCK**. It is connected to **TO 40**.
- VAR PULSE GEN. (46):** A variable pulse generator. It receives **MODE SEL** (3 bits) and **TO 37**. It outputs **DATA SYMBOLS** (9 bits, labeled **LSB** and **MSB**) and a **DIT CLOCK**. It is connected to **TO 34** and **TO 36**.
- 1x9 PAR IN SER OUT (14):** A register with **RD** (Read) and **WR** (Write) ports. It receives **DATA SYMBOLS** and **DIT CLOCK**. It outputs **8 DATA** and is connected to **TO 34** and **TO 36**.
- DIVIDER (~8) (50):** A divider block with an **R** (Reset) port. It receives **8 DATA** and outputs **8 DATA**. It is connected to **TO 34** and **TO 36**.
- Fsync DELAY (~208) (52):** A delay block with **CL** (Clock) and **R** (Reset) ports. It receives **Fsync** and **TO 34**. It outputs **RS BLOCK**, **DECIN**, and **Fsync** to **TO 42**.

Inputs and Outputs:

- Inputs:** **MODE SEL** (3 bits), **TO 37**, **TO 34** (8xSYMBOL CLOCK), **TO 36** (9 DATA SYMBOLS), **TO 34** (SYMBOL CLOCK), **TO 34** (Fsync).
- Outputs:** **8 DATA**, **BYTE CLOCK**, **TO 40**, **TO 42** (RS BLOCK, DECIN, Fsync).

$B=4, N=8$

	1	2	3	4
1	0	7	3	1
2	1	8	4	2
3	2	9	5	3
4	3	10	6	4
5	4	11	7	5
6	5	12	8	6
7	6	0	9	7
8	7	1	10	8
9	8	2	11	9
10	9	3	12	10
11	10	4	0	11
12	11	5	1	12
13	12	6	2	0

[illegible]

BNSDOCID: <WO__9518494A1_1_>

FIG. 8

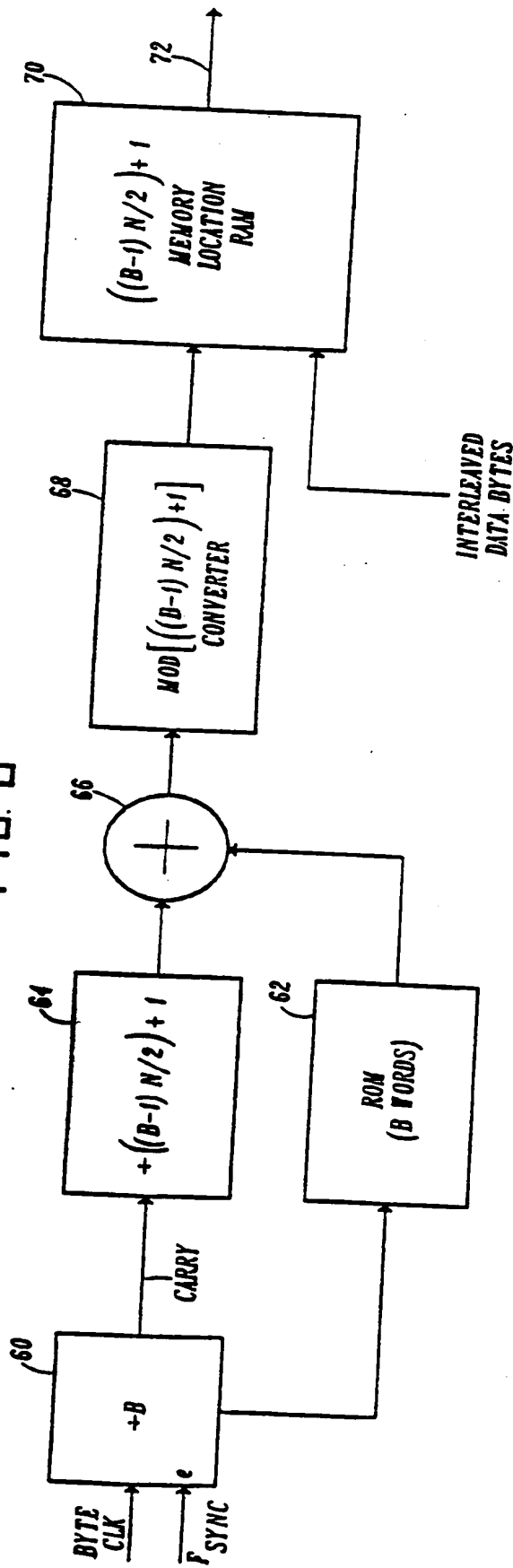
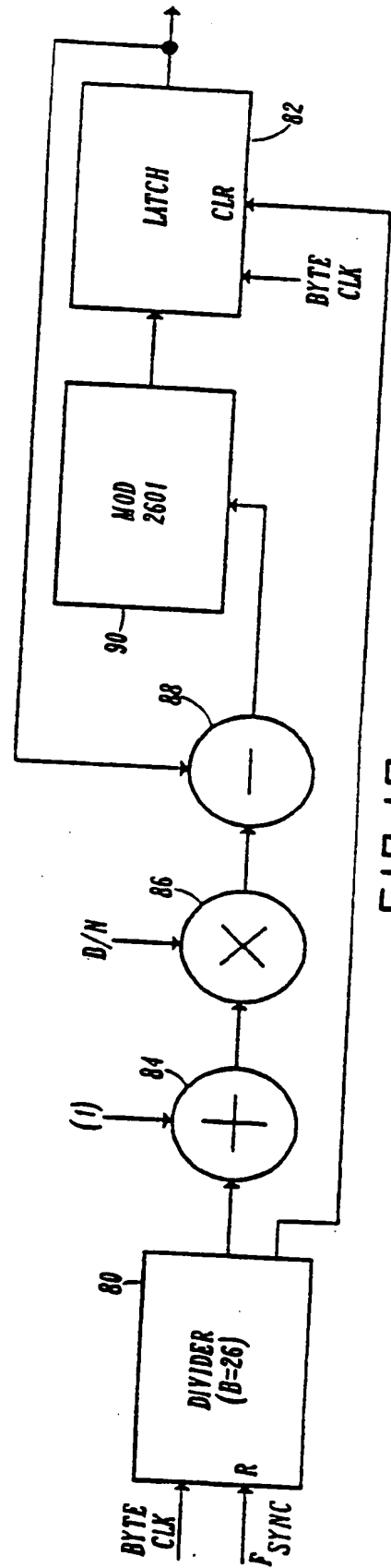


FIG. 10



INTERNATIONAL SEARCH REPORT

International Application No.
PCT/US 94/14394

A. CLASSIFICATION OF SUBJECT MATTER

IPC 6- H04L1/12 H04L25/02 H04L25/49 H03M5/20 H04N11/00
H04N7/24 H04N7/66 H04L27/34

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
IPC 6 H04L H03M H04N

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practical, search terms used)

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	<p>WO,A,92 22162 (BRITISH TELECOMMUNICATIONS PUBLIC LIMITED COMPANY) 10 December 1992 see abstract; claims; figures see page 2, line 1 - page 4, line 6 see page 5, line 16 - line 27 see page 8, line 15 - line 18 see page 9, line 10 - line 12</p> <p style="text-align: center;">--- -/-</p>	1-13

☒ Further documents are listed in the continuation of box C.

☒ Patent family members are listed in annex.

* Special categories of cited documents:

- "A" document defining the general state of the art which is not considered to be of particular relevance
- "B" earlier document but published on or after the international filing date
- "L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)
- "O" document referring to an oral disclosure, use, exhibition or other means
- "P" document published prior to the international filing date but later than the priority date claimed

- "T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
- "X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
- "Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art
- "A" document member of the same patent family

Date of the actual completion of the international search

6 April 1995

Date of mailing of the international search report

02.05.95

Name and mailing address of the ISA

European Patent Office, P.B. 5818 Patentlaan 2
NL - 2280 HV Rijswijk
Tel. (+31-70) 340-2040, Tx. 31 651 epo nl,
Fax (+31-70) 340-3016

Authorized officer

Gries, T

INTERNATIONAL SEARCH REPORT

Internal Application No
PCT/US 94/14394

C.(Continuation) DOCUMENTS CONSIDERED TO BE RELEVANT		
Category	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	<p>ELECTRONICS AND COMMUNICATION ENGINEERING JOURNAL, vol.4, no.4, August 1992, LONDON GB pages 167 - 176, XP307895 W. T. WEBB: 'QAM: the modulation scheme for future mobile radio communications ?' see page 169, middle column, paragraph 2 - page 171, left column, paragraph 1; figures 3-5 see page 172, right column, last paragraph - page 174, right column, line 2; figures 8-11</p>	1-13
A	<p>WO,A,93 14588 (QUALCOMM INCORPORATED) 22 July 1993 see claims; figures 1,2,4 see page 1, line 29 - page 2, line 13 see page 6, line 18 - line 20</p>	1-13
A	<p>EP,A,0 050 082 (SANGAMO WESTON INC.) 21 April 1982 see abstract; claims; figures 2,6-10 see page 2, line 16 - page 3, line 29</p>	1,7
A	<p>US,A,5 087 975 (CITTA ET AL.) 11 February 1992 cited in the application see abstract; figures</p>	
P,X	<p>EP,A,0 584 865 (PHILIPS ELECTRONICS UK LIMITED) 2 March 1994 see abstract; claims 1,4,5; figures see page 2, line 18 - page 3, line 22 see page 5, line 11 - page 6, line 5</p>	1-13

Form PCT/ISA/210 (continuation of second sheet) (July 1992)

INTERNATIONAL SEARCH REPORT

Information on patent family members

International Application No
PCT/US 94/14394

Patent documents cited in search report	Publication date	Patent family member(s)	Publication date
WO-A-9222162	10-12-92	AU-B- 656972 AU-A- 1772492 CA-A- 2110578 EP-A- 0587620 JP-T- 6507763	23-02-95 08-01-93 10-12-92 23-03-94 01-09-94
WO-A-9314588	22-07-93	AU-B- 3476793 CA-A- 2128327 EP-A- 0621998 FI-A- 943410 NO-A- 942670	03-08-93 22-07-93 02-11-94 16-09-94 16-09-94
EP-A-0050082	21-04-82	US-A- 4355397	19-10-82
US-A-5087975	11-02-92	EP-A- 0556306 JP-T- 6501600 WO-A- 9209174	25-08-93 17-02-94 29-05-92
EP-A-0584865	02-03-94	JP-A- 6177846	24-06-94

Form PCT/ISA/238 (patent family member) (July 1992)